



INTEGO
GROUP

Big Data Sets Seen as a Big Problem and How to Deal with Them

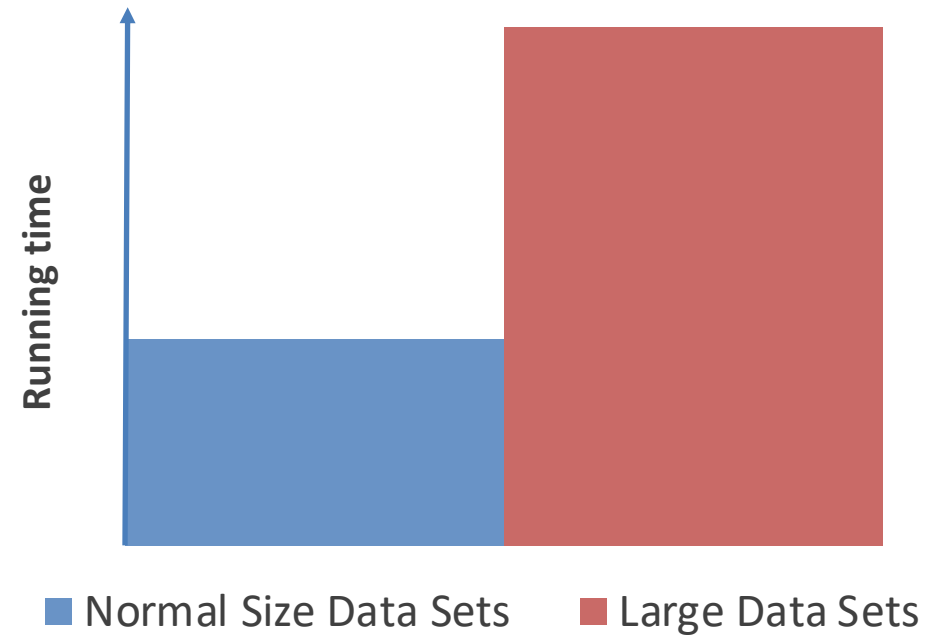
Daniil Shliakhov, Kharkiv, Ukraine



Frankfurt 2018



Running time is an issue!



Parameter	Treatment	n	Mean	SD	Median	Min	Max
Alkaline Phosphatase (U/L)	Baseline						
	Pooled TRT1	xxx	xxx.x	xxx.xx	xxx.xx	xxx	xxx
	Pooled TRT2	xxx	xxx.x	xxx.xx	xxx.xx	xxx	xxx
	Cycle 1						
	Pooled TRT1	xxx	xxx.x	xxx.xx	xxx.xx	xxx	xxx
	Pooled TRT2	xxx	xxx.x	xxx.xx	xxx.xx	xxx	xxx
	Cycle 2						
	Pooled TRT1	xxx	xxx.x	xxx.xx	xxx.xx	xxx	xxx
	Pooled TRT2	xxx	xxx.x	xxx.xx	xxx.xx	xxx	xxx



GENERAL TIPS
RETRIEVING SAS DATA SETS
VIEW option



A simple data step?

How much time it may take to run this step if ADAM.ADLB is huge?

```
data adlb;  
    set adam.adlb;  
run;
```

35 seconds? Too loooooong 😞

```
data adlb;  
    set adam.adlb;  
run;
```

Real time	34:23
CPU time	3:10

A SAS VIEW is a type of SAS data set that retrieves data values from other files

```
data adlb / VIEW=adlb;  
    set adam.adlb;  
run;
```

Less than 1 second? MAGIC!

```
data adlb / VIEW=adlb;  
    set adam.adlb;  
run;
```

Real time	0:03
CPU time	0:01

Simple merge, common sort... How long?

```
data analysis;  
    merge adam.adsl adam.adlb;  
    by studyid usubjid;  
run;
```

```
proc sort data=analysis;  
    by trt01an parcat paramcd avisitn;  
run;
```

Simple merge, common sort... Too loooooong again

```
data analysis;  
    merge adam.adsl adam.adlb;  
    by studyid usubjid;  
run;  
  
proc sort data=analysis;  
    by trt01an parcat paramcd avisitn;  
run;
```

	Data Step	PROC step
Real time	36:04	1:12.40
CPU time	8:18	12.45

Adding VIEW option to do magic

```
data analysis / VIEW=analysis;  
    merge adam.adsl adam.adlb;  
    by studyid usubjid;  
run;
```

```
proc sort data=analysis out=analysis_sorted;  
    by trt01an parcat paramcd avisitn;  
run;
```

VIEW option gives extra time to drink coffee with colleagues

```
data analysis / VIEW=analysis;
  merge adam.adsl adam.adlb;
  by studyid usubjid;
run;
```

```
proc sort data=analysis out=analysis_sorted;
  by trt01an parcat paramcd avisitn;
run;
```

	Data Step	PROC step
Real time	0:04	1:17.65
CPU time	0:02	20.32



GENERAL TIPS
RETRIEVING SAS DATA SETS
IF or WHERE?



IF vs. WHERE. Who is the champion?

```
data adlb;  
    set adam.adlb;  
    if ANL01FL = 'Y';  
run;
```

```
data adlb;  
    set adam.adlb;  
    where ANL01FL = 'Y';  
run;
```

IF is champion! Woohoo!

```
data adlb;  
    set adam.adlb;  
    if ANL01FL = 'Y';  
run;
```

```
data adlb;  
    set adam.adlb;  
    where ANL01FL = 'Y';  
run;
```

	<u>IF statement</u>	WHERE statement
Real time	31:64	33:31
CPU time	3:53	5:68

INDEX helps WHERE to win 😊

```
data adlb;
  set adam.adlb;
  if ANL01FL = 'Y';
run;
```

```
data adlb;
  set adam.adlb;
  where ANL01FL = 'Y';
run;
```

INDEX APPLIED

	IF statement	<u>WHERE</u> <u>statement</u>
Real time	32:15	27:26
CPU time	4:28	2.98



SAS PROCEDURES **DESCRIPTIVE STATS**



```
proc means data=adlb noprint;  
  by trt01an parcat paramcd avisitn;  
  var aval;  
  output out = mnout  
         n = n  
         mean = mean  
         median = median  
         std = std  
         min = min  
         max = max;  
  
run;
```

```
proc univariate data=adlb noprint;  
  by trt01an parcat paramcd avisitn;  
  var aval;  
  output out = mnout  
         n = n  
         mean = mean  
         median = median  
         std = std  
         min = min  
         max = max;  
  
run;
```

```
proc summary data=adlb noprint;  
  by trt01an parcat paramcd avisitn;  
  var aval;  
  output out = mnout  
         n = n  
         mean = mean  
         median = median  
         std = std  
         min = min  
         max = max;  
  
run;
```

```
proc sql noprint;
    create table mnout as
    select trt01an, parcat, paramcd, avisitn,
           COUNT(*) as n,
           MEAN(aval) as mean,
           MEDIAN(aval) as median,
           STD(aval) as std,
           MIN(aval) as min,
           MAX(aval) as max
    from adlb
    group by trt01an, parcat, paramcd, avisitn
quit;
```

DESCRIPTIVE STATS COMPARISON

	MEANS	UNIVARIATE	<u>SUMMARY</u>	SQL
Real time	15:14	24:78	13:24	13:45
CPU time	3:38	1:76	3:33	2:53



SAS PROCEDURES **FREQUENCY**



```
proc freq data=adlb noprint;  
    by trt01an parcat paramcd;  
    tables avisitn / out=frouit;  
run;
```

```
proc summary data=adlb nway noprint;  
    by trt01an parcat paramcd avisitn;  
    output out=frouit;  
run;
```

```
proc sql noprint;  
    create table frout as  
    select trt01an, parcat, paramcd, avisitn,  
           COUNT(*) as count  
    from adlb  
    group by trt01an, parcat, paramcd, avisitn  
quit;
```

FREQUENCY COMPARISON

	FREQ	SQL	<u>SUMMARY</u>
Real time	13:62	12:19	12:02
CPU time	2:04	1:63	0:88



CONCLUSIONS

Do not be afraid to work with big data sets.

Just choose the “right” procedure!





INTEGO
GROUP

THANK YOU

Daniil Shliakhov
daniil.shlyakhov@intego-group.com
Kharkiv, Ukraine
www.intego-group.com

